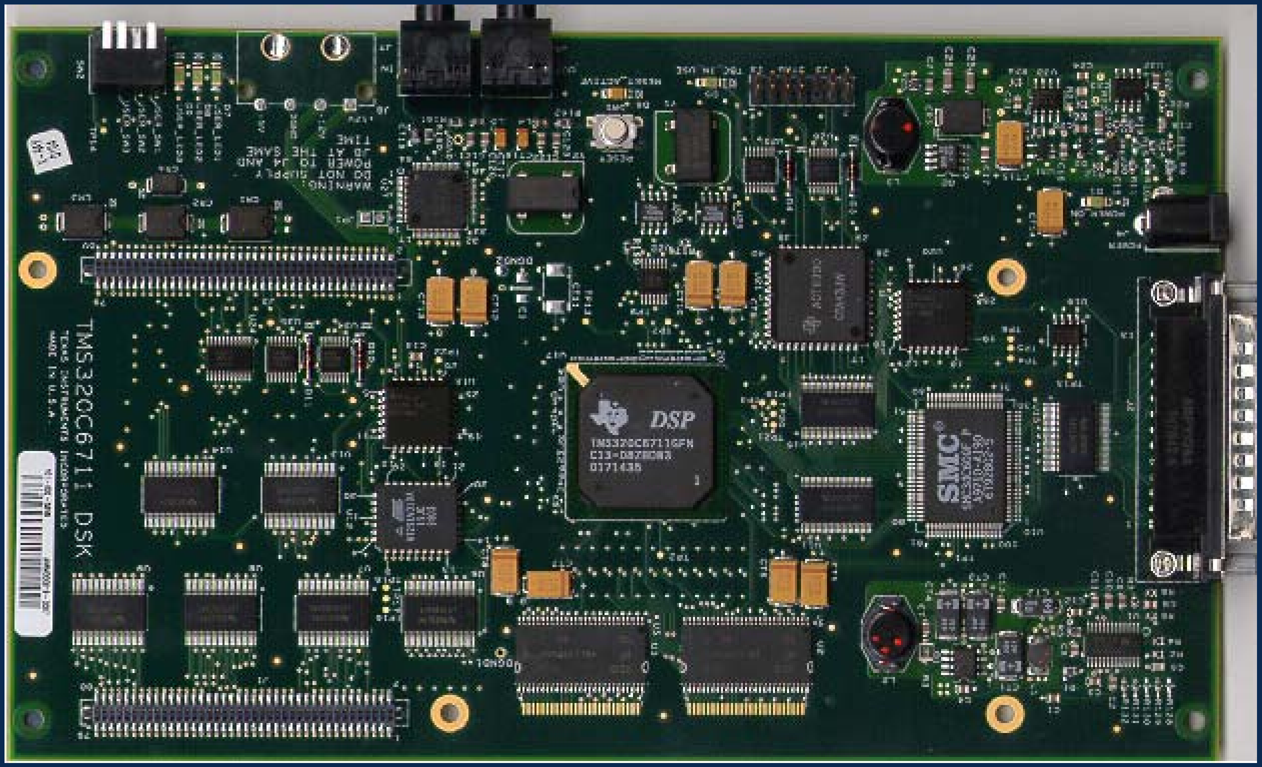
TECHNISCHE UNIVERSITÄT DRESDEN

Vodafone Chair Mobile Communications Systems, Prof. Dr.-Ing. G. Fettweis

vodafone chair

Digital Signal Transmission Lab
SS 08

Oliver Arnold
Steffen Kunze



Introduction

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

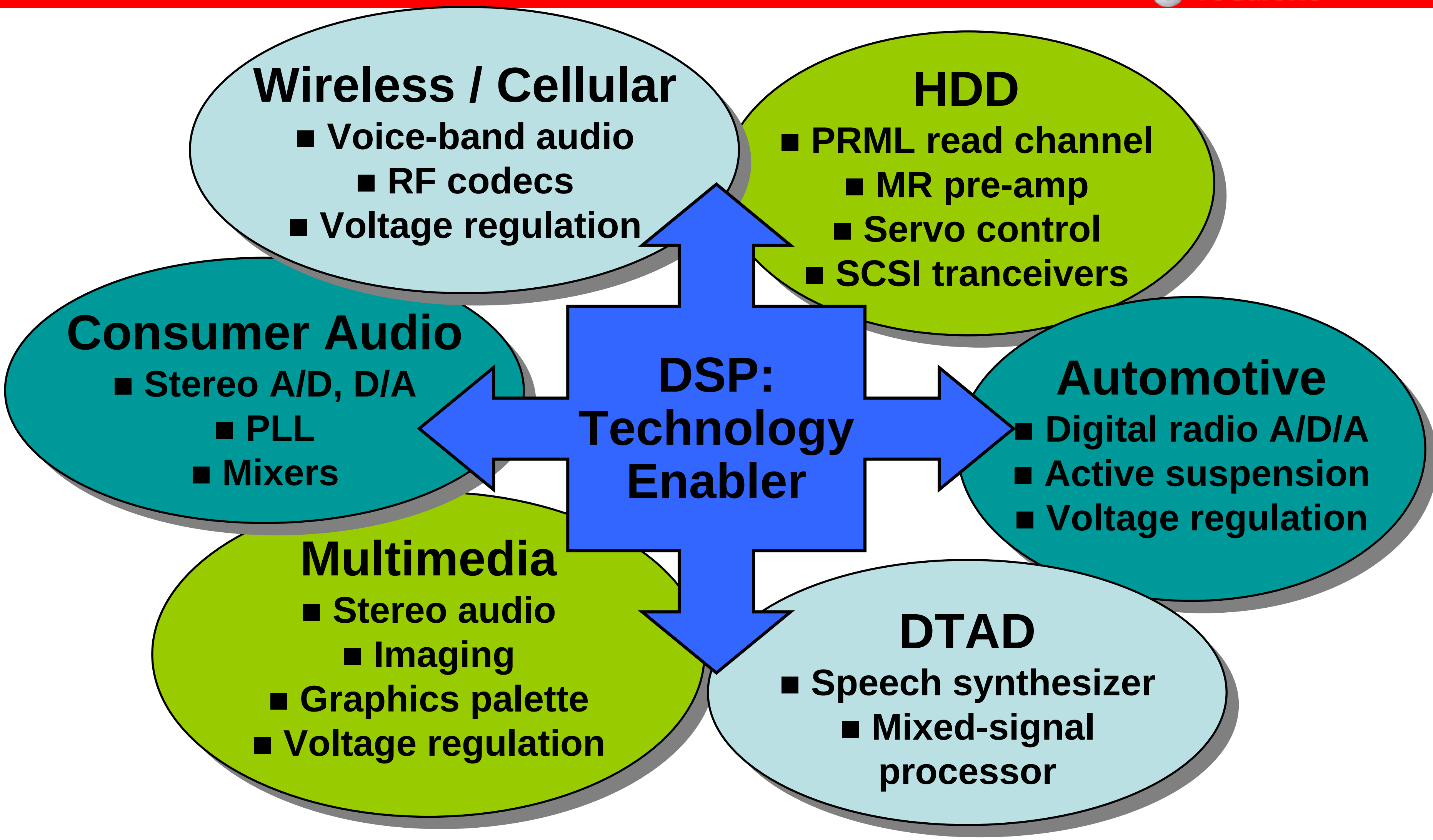
- **Hardware**
  - Why to use digital signal processing?
  - General introduction to DSPs
  - The TMS320C6711 DSP
    - Architecture Overview
    - Peripherals
- **DSK6711 evaluation board - Software**
  - Code Composer Studio
  - DSP/BIOS
  - Multi-channel Buffered Serial Port (McBSP)

Hardware

Digital Signal Processing (DSP)

TECHNISCHE UNIVERSITÄT DRESDEN

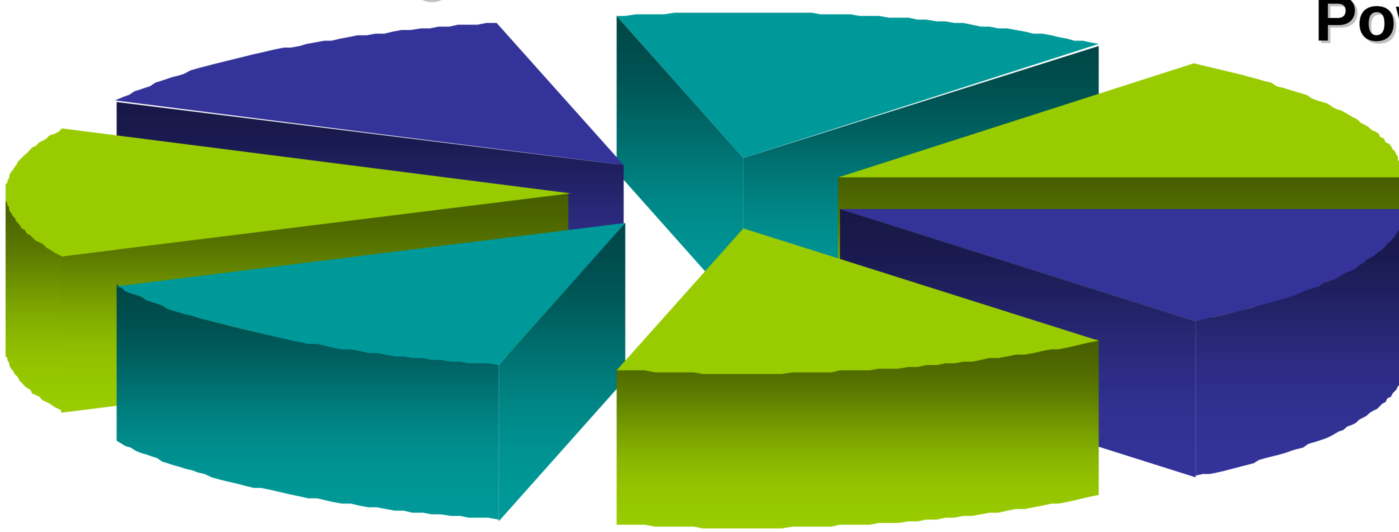
vodafone chair



System Considerations

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair



Ease-of Use
- Programming
- Interfacing
- Debugging

Cost
- Device cost
- System cost
- Development cost
- Time to market

Integration
- Memory
- Peripherals

Why Go Digital?

- Digital signal processing techniques are now so **powerful** that sometimes it is extremely difficult, if not impossible, for analogue signal processing to achieve similar performance.
- **Examples**:
  - FIR filter with linear phase
  - Adaptive filters

Why Go Digital?

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

- Analogue signal processing is achieved by using **analogue components** such as:
  - Resistors
  - Capacitors
  - Inductors
- The inherent **tolerances** associated with these components, temperature, voltage changes and mechanical vibrations can dramatically affect the effectiveness of the analogue circuitry

Why Go Digital?

- **With DSP? - It is easy to:**
  - Change applications
  - Correct applications
  - Update applications
- **Additionally DSPs reduce:**
  - Noise susceptibility
  - Chip count
  - Development time
  - Cost
  - Power consumption

General Introduction to DSPs

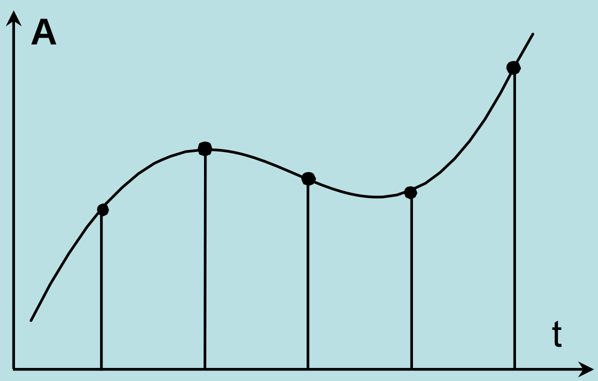
What Problem Are We Trying To Solve?

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair



Digital sampling of an analog signal:



Most DSP algorithms can be expressed as:

$$Y = \sum_{i=1}^{count} a_i * x_i$$

```
for (i = 1; i < count; i++){
  sum += m[i] * n[i]; }
```

What are the typical DSP algorithms?

- The Sum of Products (SOP) is the key element in most DSP algorithms:

Algorithm	**Equation**
Finite Impulse Response Filter	$y(n) = \sum_{k=0}^{M} a_k\, x(n-k)$
Infinite Impulse Response Filter	$y(n)=\sum_{k=0}^{M} a_k\, x(n-k)+\sum_{k=1}^{N} b_k\, y(n-k)$
Convolution	$y(n)=\sum_{k=0}^{N} x(k)h(n-k)$
Discrete Fourier Transform	$X(k) = \sum_{n=0}^{N-1} x(n)\exp[-j(2\pi / N)nk]$
Discrete Cosine Transform	$F(u)=\sum_{x=0}^{N-1} c(u).f(x).\cos\left[\frac{\pi}{2N}u(2x+1)\right]$

Why do we need DSP processors?

- Use a DSP processor when the following are required:
  - Cost saving
  - Smaller size
  - Low power consumption
  - Processing of many “high” frequency signals in real-time
- Use a GPP processor when the following are required:
  - Large memory
  - Advanced operating systems

Hardware vs. Microcode multiplication

- DSP processors are optimized to perform multiplication and addition operations.
- Multiplication and addition are done in hardware and in one cycle.
- Example: 4-bit multiply (unsigned).

```
Hardware              Microcode

    1011                   1011
  x 1110                 x 1110
 --------               --------
10011010                   0000    Cycle 1
                          1011.    Cycle 2
                         1011..    Cycle 3
                        1011...    Cycle 4
                        --------
                       10011010    Cycle 5
```

General Purpose DSP vs. DSP in ASIC

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

- Application Specific Integrated Circuits (**ASICs**) are semiconductors designed for **dedicated functions**.
- The advantages and disadvantages of using **ASICs** are listed below:

Advantages	**Disadvantages**
• High throughput • Lower silicon area • Lower power consumption • Improved reliability • Reduction in system noise • Low overall system cost	• High investment cost • Less flexibility • Long time from design to market

Floating vs. Fixed point processors

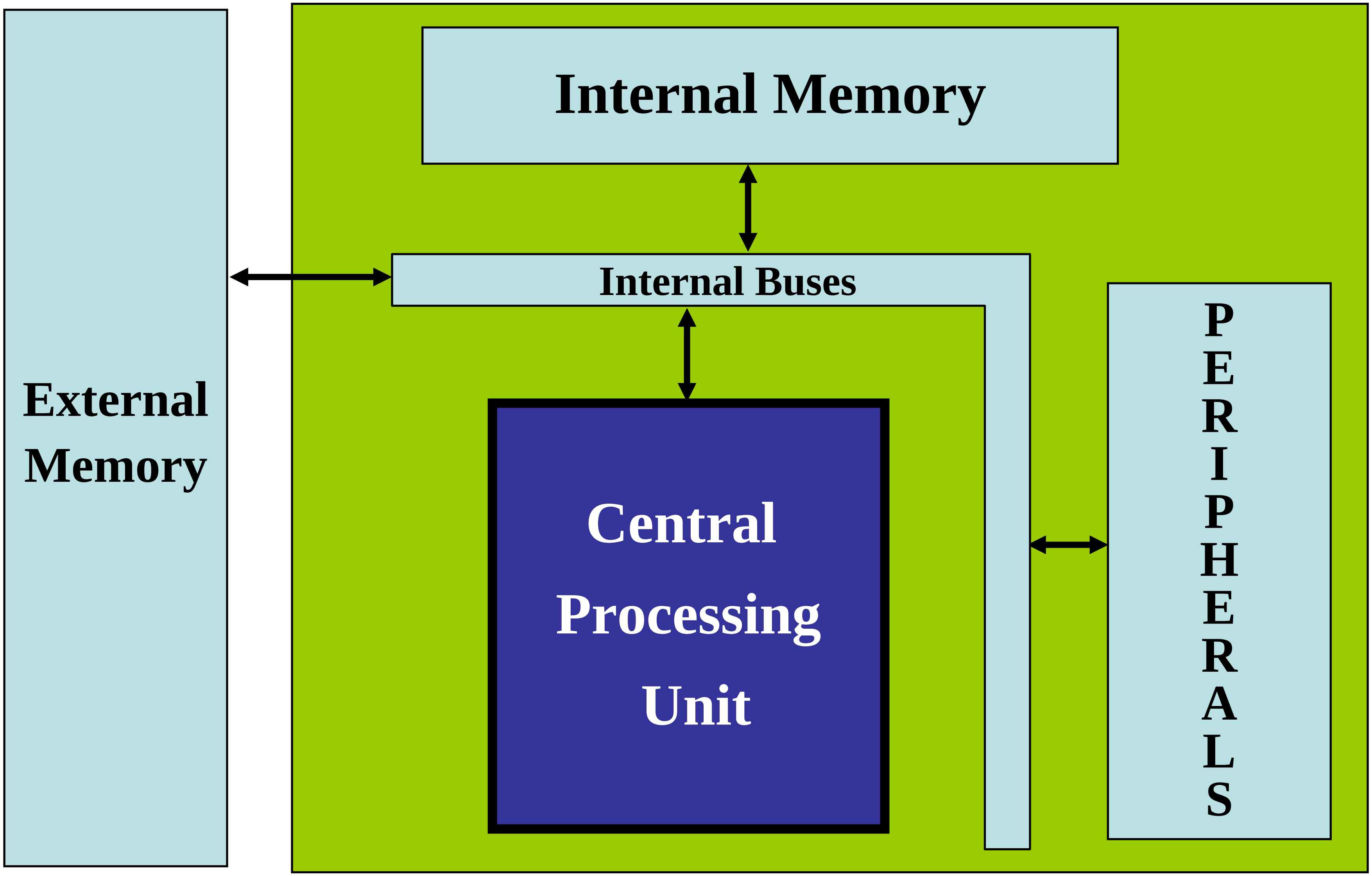
- **Applications which require:**
  - High precision
  - Wide dynamic range
  - High signal-to-noise ratio
  - Ease of use

→ Need a floating point processor

- **Drawback of floating point processors:**
  - Higher power consumption
  - Usually higher cost
  - Usually slower than fixed-point counterparts and larger in size

TMS320C6711 Architectural Overview

General DSP System Block Diagram



'6711 CPU Overview

- **Specification**
  - Clock Rate: 100/150 MHz ➔ 600/900 MFLOPS
  - 0.18-µm/5-Level Metal Process – CMOS Technology
- CPU has got **two Datapaths**, altogether:
  - Four ALUs (Floating- and Fixed-Point)
  - Two ALUs (Fixed-Point)
  - Two Multipliers (Floating- and Fixed-Point)
  - Load-Store Architecture
  - 2*16 32-Bit General-Purpose Registers

‘6711 CPU Overview

TECHNISCHE UNIVERSITÄT DRESDEN

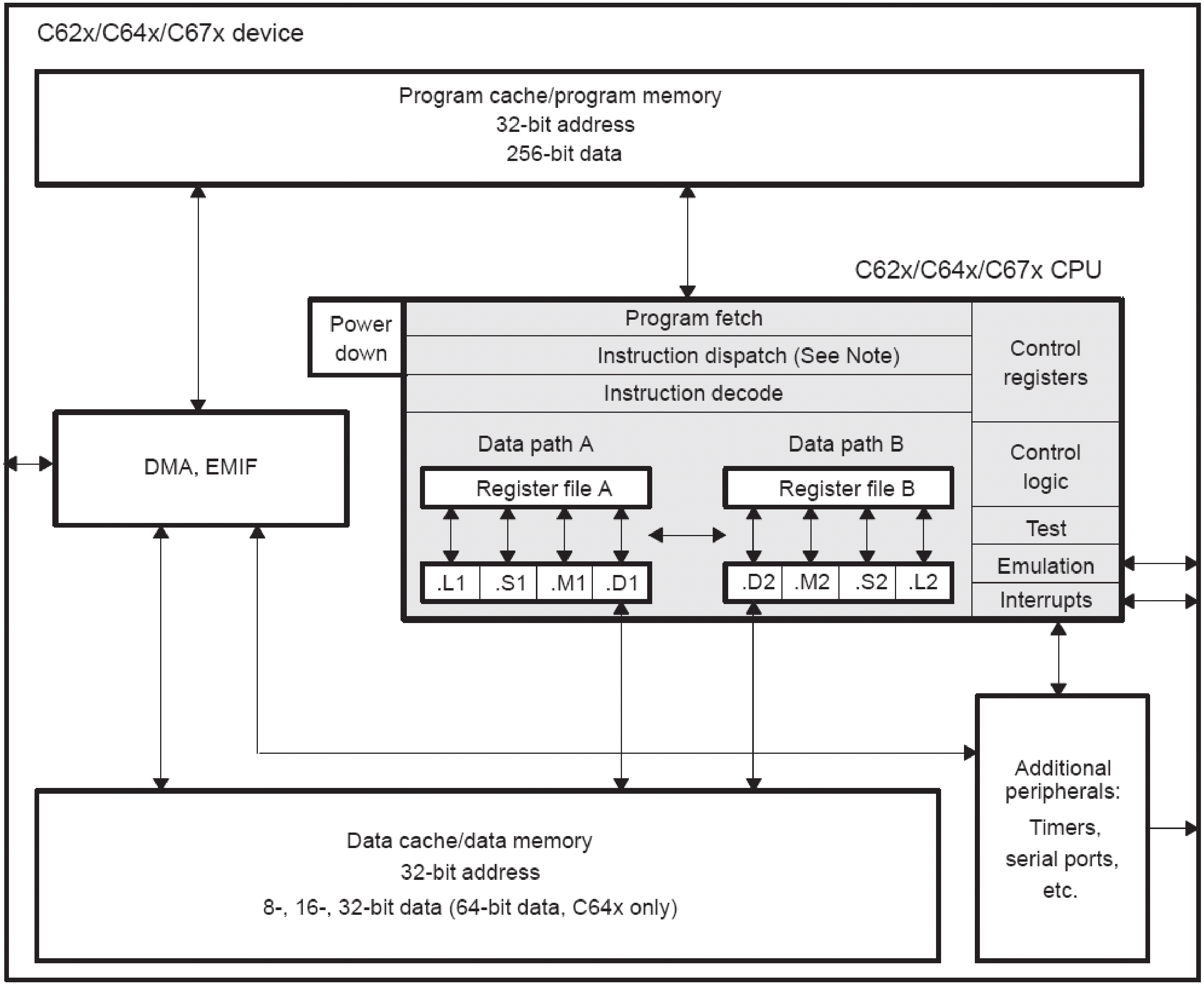
vodafone chair

- VelociTI ➔ advanced very-long instruction words (**VLIW**)
  - Program Memory Width is 256 Bit
  - Up to 8 32-Bit instructions can be executed in parallel/Cycle
  - 16, 32 and 40 bit fixed point operands
  - 32 and 64 bit floating point operands
  - Instruction parallelism is detected at compile-time
    - no data dependency checking is done in Hardware.
  - Instruction Packing Reduces Code Size
  - All operations work on registers

Memory Architecture

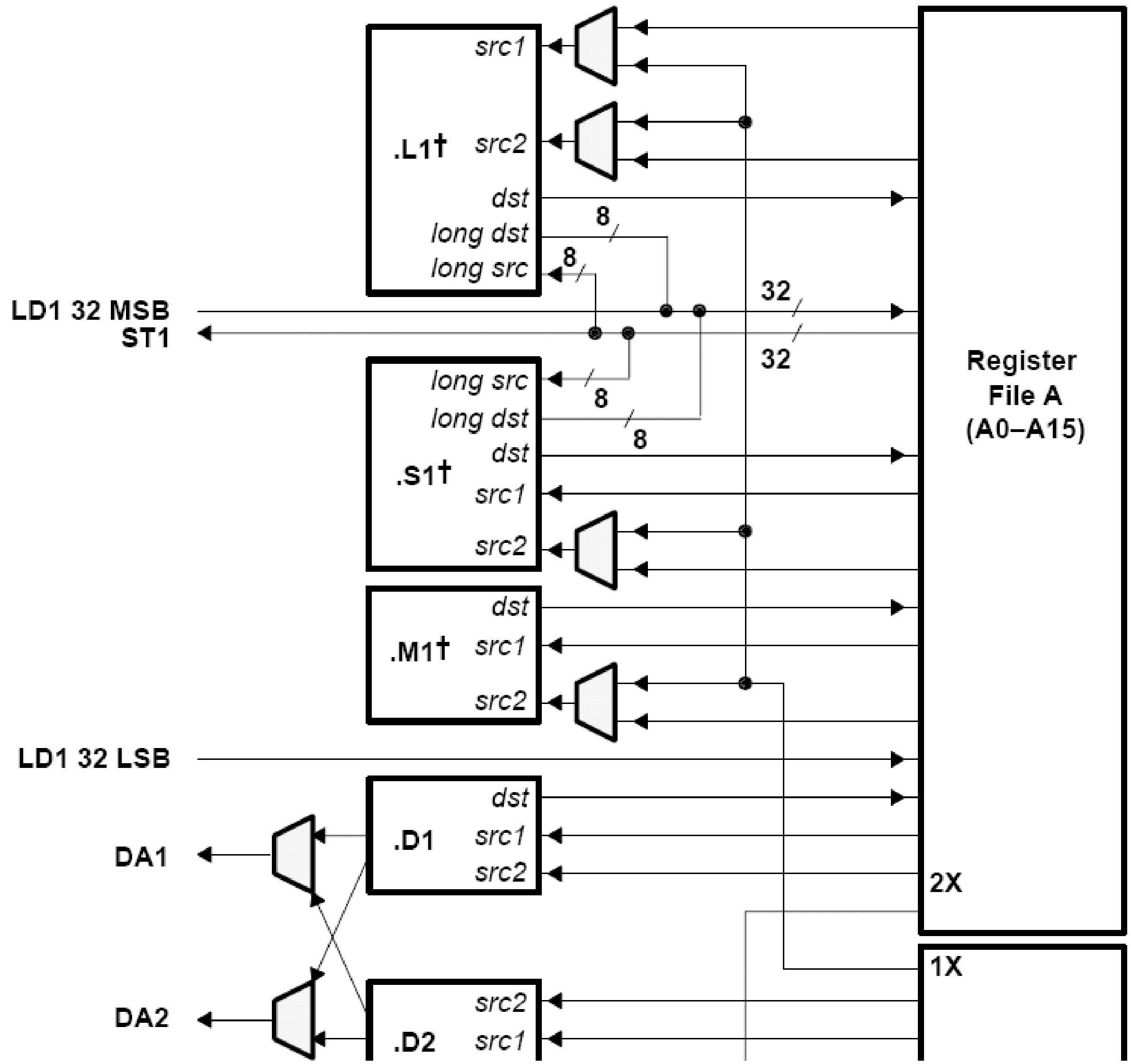
- 4K-Byte L1P Program Cache (Direct Mapped)
- 4K-Byte L1D Data Cache (2-Way Set-Associative)
- 64K-Byte L2 Unified Mapped RAM/L2 Cache (Flexible Data/Program Allocation)

Functional Block and CPU Diagram



A ‘6711 Datapath

- .S & .L
  - Arithmetic, Logical & Branch functions
- .M
  - Multiply, Rotation, Bit expansion
- .D
  - Data-addressing
  - Only way to access memory
- Cross path



Functional Units and Operations Performed

Functional Unit	Fixed-Point Operations	Floating-Point Operations
.L unit (.L1,.L2)	32/40-bit arithmetic and compare operations Leftmost 1 or 0 bit counting for 32 bits Normalization count for 32 and 40 bits 32-bit logical operations	Arithmetic operations Conversion operations: DP $\rightarrow$ SP, INT $\rightarrow$ DP, INT $\rightarrow$ SP
.S unit (.S1, .S2)	32-bit arithmetic operations 32/40-bit shifts and 32-bit bit-field operations 32-bit logical operations Branches Constant generation Register transfers to/from the control register file (.S2 only)	Compare reciprocal and reciprocal square-root operations Absolute value operations SP to DP conversion operations
.M unit (.M1, .M2)	16 $\times$ 16 bit multiply operations	32 $\times$ 32 bit multiply operations Floating-point multiply operations
.D unit (.D1, .D2)	32-bit add, subtract, linear and circular address calculation Loads and stores with a 5-bit constant offset Loads and stores with a 15-bit constant offset (.D2 only)	Load double word with a 5-bit constant offset

C6700: Instruction Set

.S
.L
.D
.M

.S Unit

ADD	NEG	ABSSP
ADDK	NOT	ABSDP
ADD2	OR	CMPGTSP
AND	SET	CMPEQSP
B	SHL	CMPLTSP
CLR	SHR	CMPGTDP
EXT	SSHL	CMPEQDP
MV	SUB	CMPLTDP
MVC	SUB2	RCPSP
MVK	XOR	RCPDP
MVKH	ZERO	RSQRSP
		RSQRDP
		SPDP

.D Unit

ADD		NEG	
ADDAB	(B/H/W)	STB	(B/H/W)
LDB	(B/H/W)	SUB	
LDDW		SUBAB	(B/H/W)
MV		ZERO	

.L Unit

ABS	NOT	ADDSP
ADD	OR	ADDDP
AND	SADD	SUBSP
CMPEQ	SAT	SUBDP
CMPGT	SSUB	INTSP
CMPLT	SUB	INTDP
LMBD	SUBC	SPINT
MV	XOR	DPINT
NEG	ZERO	SPRTUNC
NORM		DPTRUNC
		DPSP

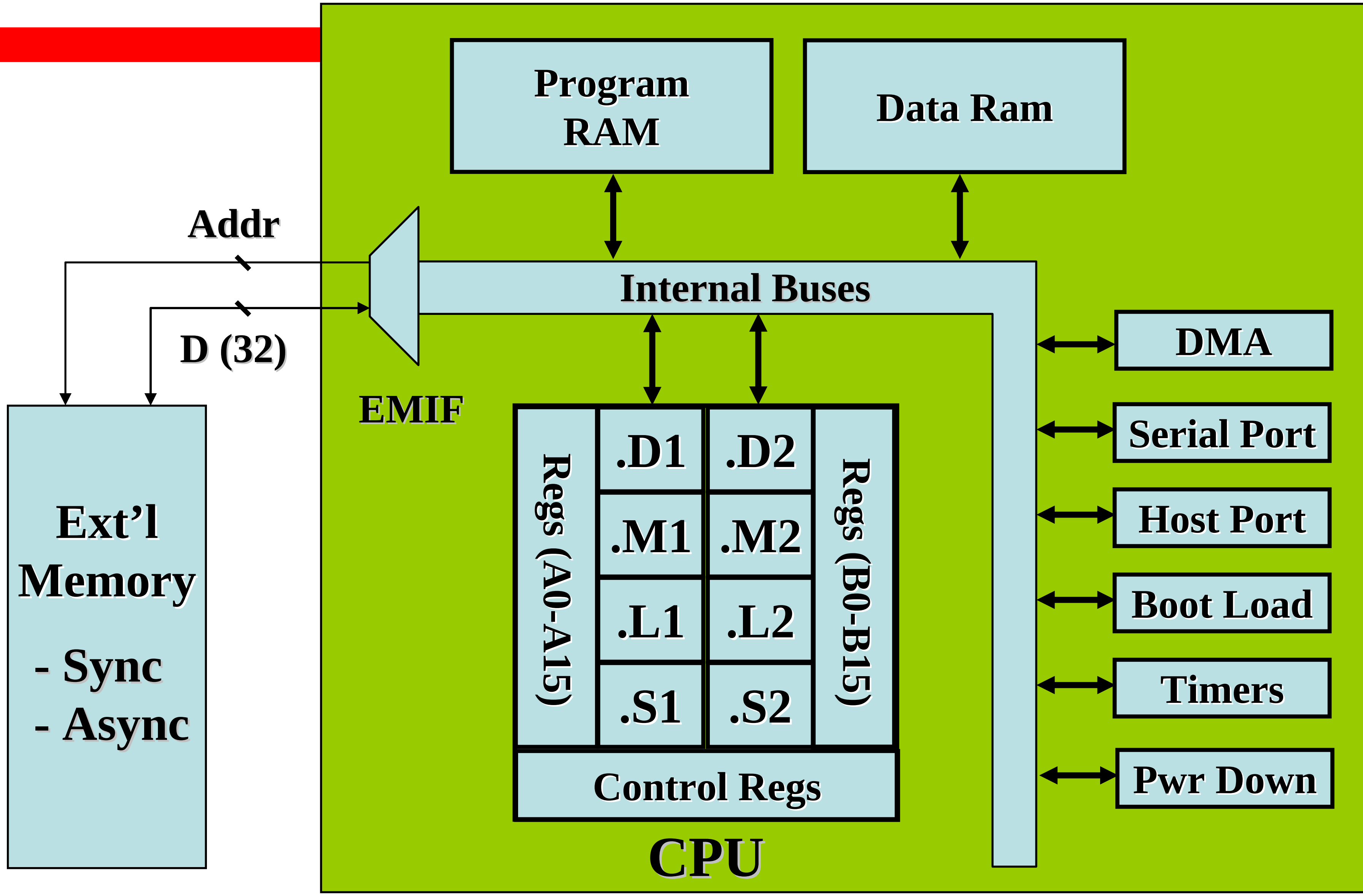
.M Unit

MPY	SMPY	MPYSP
MPYH	SMPYH	MPYDP
MPYLH		MPYI
MPYHL		MPYID

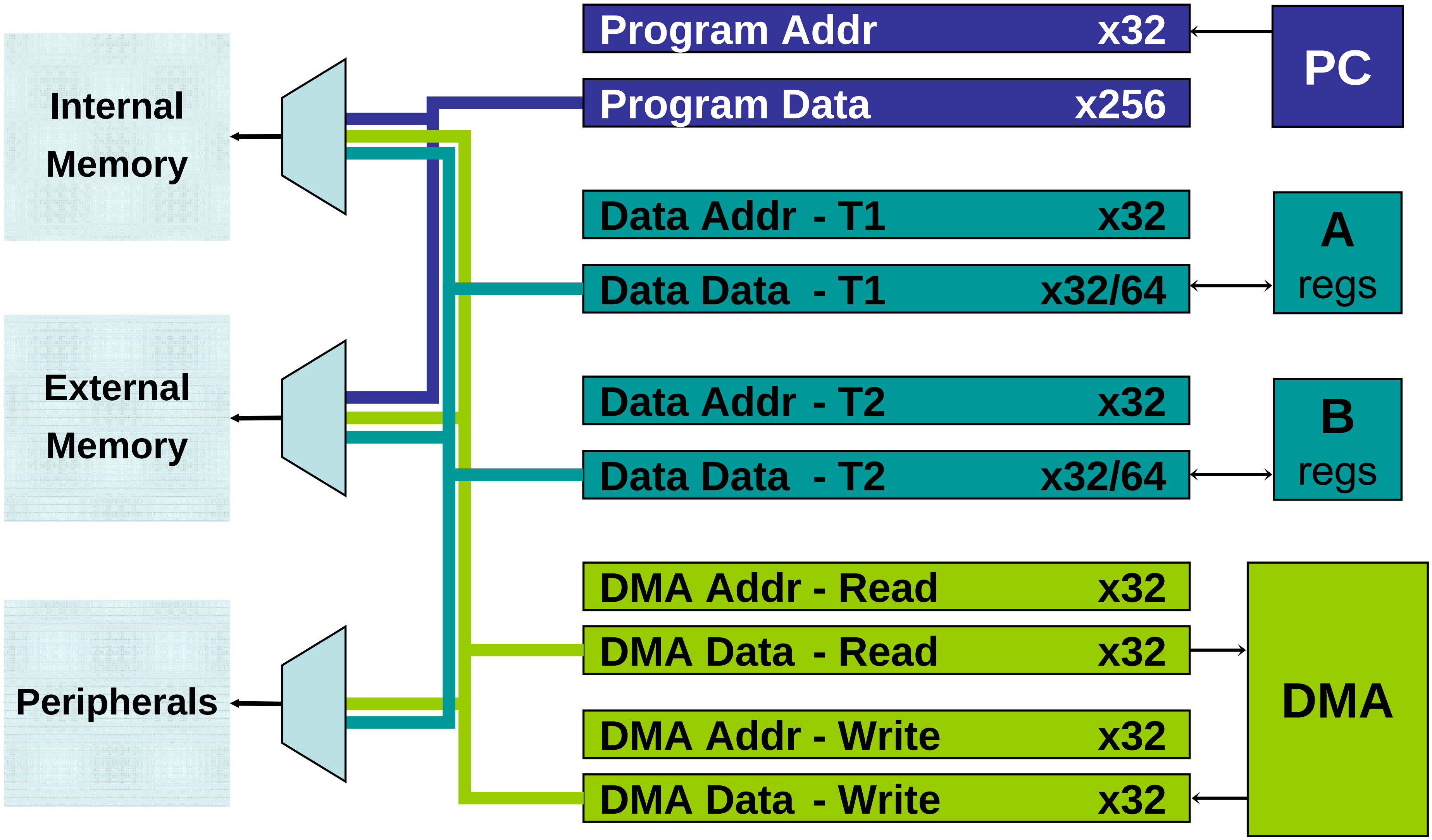
No Unit Used

NOP	IDLE

'C6x System Block Diagram



‘C6000 Internal Buses



How are Peripherals Controlled?

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

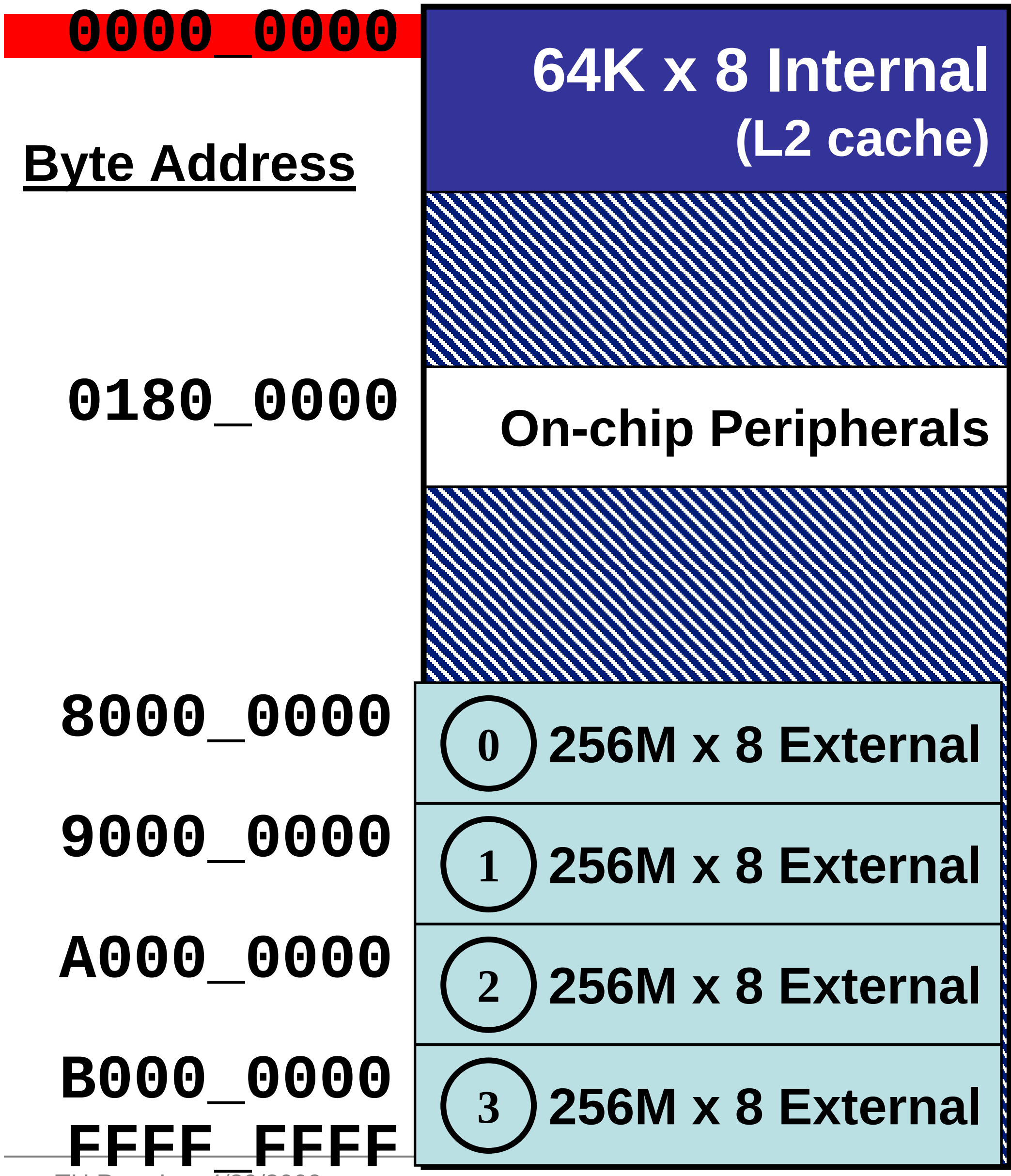
- Control and configuration of internal peripherals is done by memory mapped control registers
- There is a separate memory mapped register file of control registers

Example of Timer mode control register:

31 – 12	11	10	9	8
Rsvd	TSAT	INVIMP	CLKSRC	C/P

7	6	5	4	3	2	1	0
HLD	GO	Rsvd	PWID	DATIN	DATOUT	INVOUT	Func

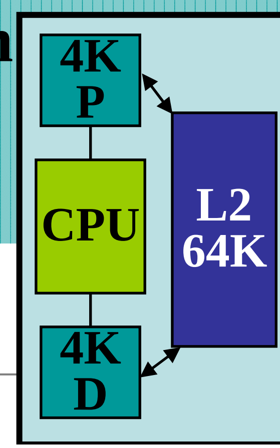
‘C6711 Memory Map



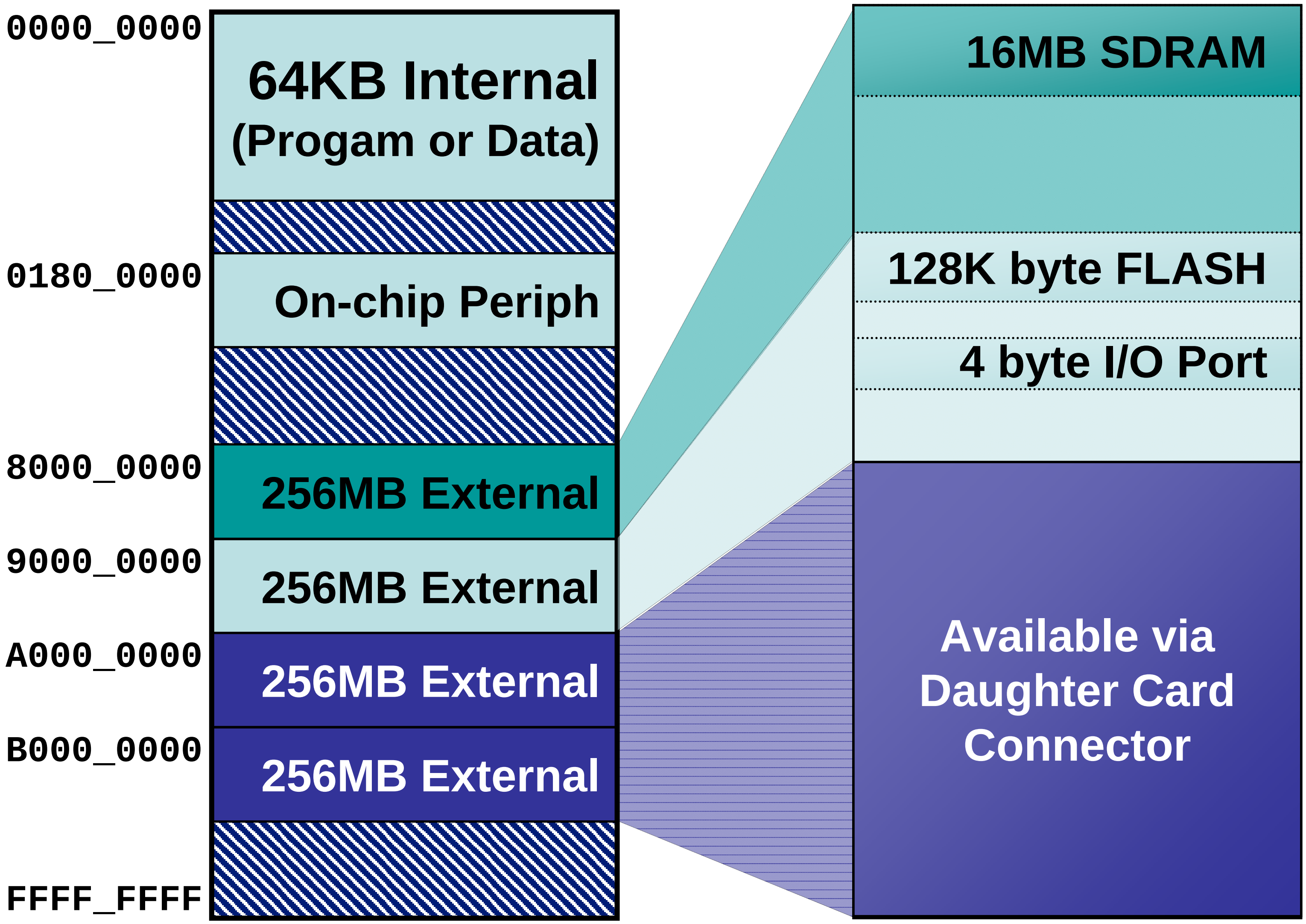
External Memory
- Async (SRAM, ROM, etc.)
- Sync (SBSRAM, SDRAM)

Internal Memory
- Unified (data or prog)
- 4 blocks - each can be RAM or cache

Level 1 Cache
- 4KB Program
- 4KB Data
- Not in map



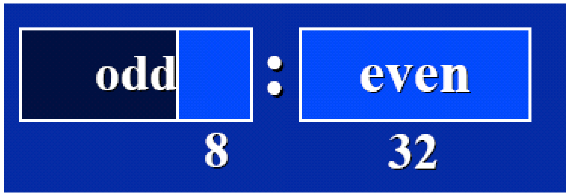
Memory Map



- LED's
- Switches
- DSK status
- DSK rev#
- Daughter Card

Operands

- **Operands can be**
  - 5-bit constants (or 16-bit in some special instruct.)
  - 32-bit Registers
  - 40-bit Registers
  - 64-bit Registers
- A 40-bit or a 64-bit register can be obtained by concatenating two registers
  - The registers must be from the same side
  - The first register must be even and the second odd (e.g. A1:A0, B9:B8 or A15:A14)
  - The registers must be consecutive



Conditional execution

vodafone chair

- All instructions in each Functional Unit of both Data paths can be executed conditionally
- Only the Registers A1, A2, B0, B1, B2 can hold the condition
- Conditional Execution uses the Syntax

[!condition] Instruction

e.g

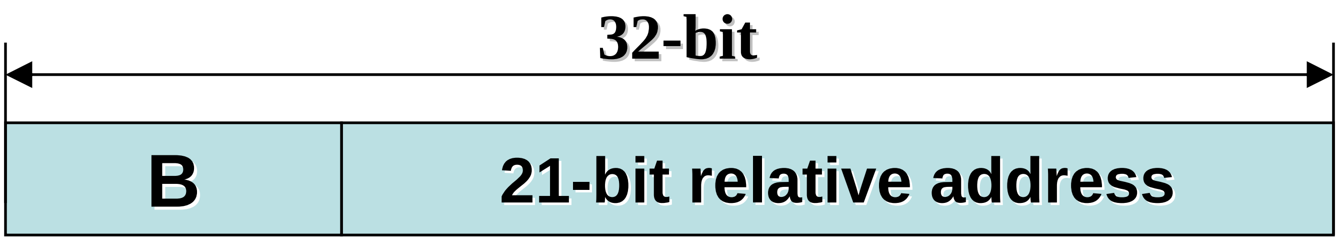
```
[!B0]   ADD.L1   A1,A2,A3   ; add if B0 ==0
[B0]    ADD.L1   A1,A2,A3   ; add if B0 != 0
```

Branches

- Branches are required to realize loops and change the program flow
- Branches are very useful in conjunction with conditional execution
- There are two branch types supported:
  - Relative Branching
  - Absolute Branching

More on the Branch Instruction (1)

- With this processor all the instructions are encoded in a 32-bit.
- Therefore the label must have a dynamic range of less than 32-bit as the instruction B has to be coded.



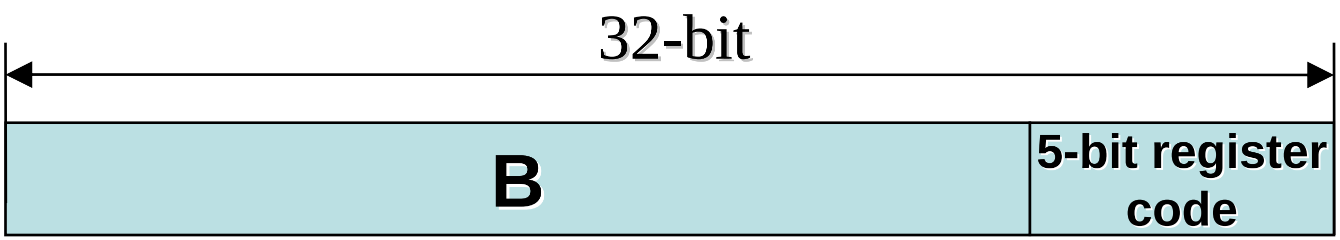
- Case 1: B .S1 *label*
  - Relative branch.
  - Label limited to **+/- 2^{20} offset**.

More on the Branch Instruction (2)

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

- By specifying a **register as an operand** instead of a label, it is possible to have an absolute branch.
- This will allow a dynamic range of 2^{32}.



- Case 2: B .S2 *register*
  - Absolute branch.
  - Operates on .S2 ONLY!

Getting Data from the Memory

- All Instructions work exclusively on Registers
- The .D Units in the Data-Paths are used to load and store the required Data from and to the Memory
- Load and Store Instructions use an Address operator X:

```
LDW.D1   X,A5
STW.D2   B11,X
```

Addressing Modes

- There are two addressing modes supported:
  - ❑ Linear Addressing
  - ❑ Circular Addressing (e.g. Convolution)
    - Circular Addressing supports block sizes 2^N
    - Only the lower N bits of the Address are modified by address arithmetic. This equals mod(2^N) operations.



- The addressing mode is selected by control register „AMR‘
- Operands for CA are limited to A4-A7, B4-B7

Floating vs. Fixed point processors

- Fixed point arithmetic
  - 16-bit (integer or fractional)
  - Signed or unsigned
- Floating point arithmetic
  - 32-bit single precision
  - 64-bit single precision
- Using signed and unsigned integers:
  - Multiplication overflow.
  - Addition overflow
    - ➔ Saturate the result
    - ➔ Double precision result
    - ➔ Fractional arithmetic

$$y(n) = \sum_{k=0}^{N-1} a(k)x(n-k)$$

e.g. If A and B are fractional then: A x B < min(A, B)

C6000 C Data Types

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

Type	Size	Representation
char, signed char	8 bits	ASCII
unsigned char	8 bits	ASCII
short	16 bits	2's complement
unsigned short	16 bits	binary
int, signed int	32 bits	2s complement
unsigned int	32 bits	binary
long, signed long	40 bits	2's complement
unsigned long	40 bits	binary
enum	32 bits	2's complement
float	32 bits	IEEE 32-bit
double	64 bits	IEEE 64-bit
long double	64 bits	IEEE 64-bit
pointers	32 bits	binary

Numerical Issues - Useful Tips

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

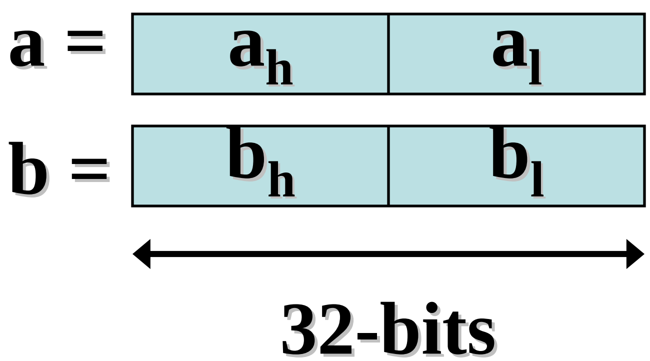
- Multiply by 2: Use shift left
- Divide by 2: Use shift right
- $Log_2 N$: Use shift
- Sine, Cosine, Log: Use look up tables
- To convert a fractional number to hex:
  - Num x 2^{15}
  - Then convert to hex

e.g: convert 0.5 to hex

- $0.5 \times 2^{15} = 16384$
- $(16384)_{dec} = (0x4000)_{hex}$

Numerical Issues - 32-bit Multiplication

- It is possible to perform 32-bit multiplication using 16-bit multipliers.
- Example: c = a x b (with 32-bit values).



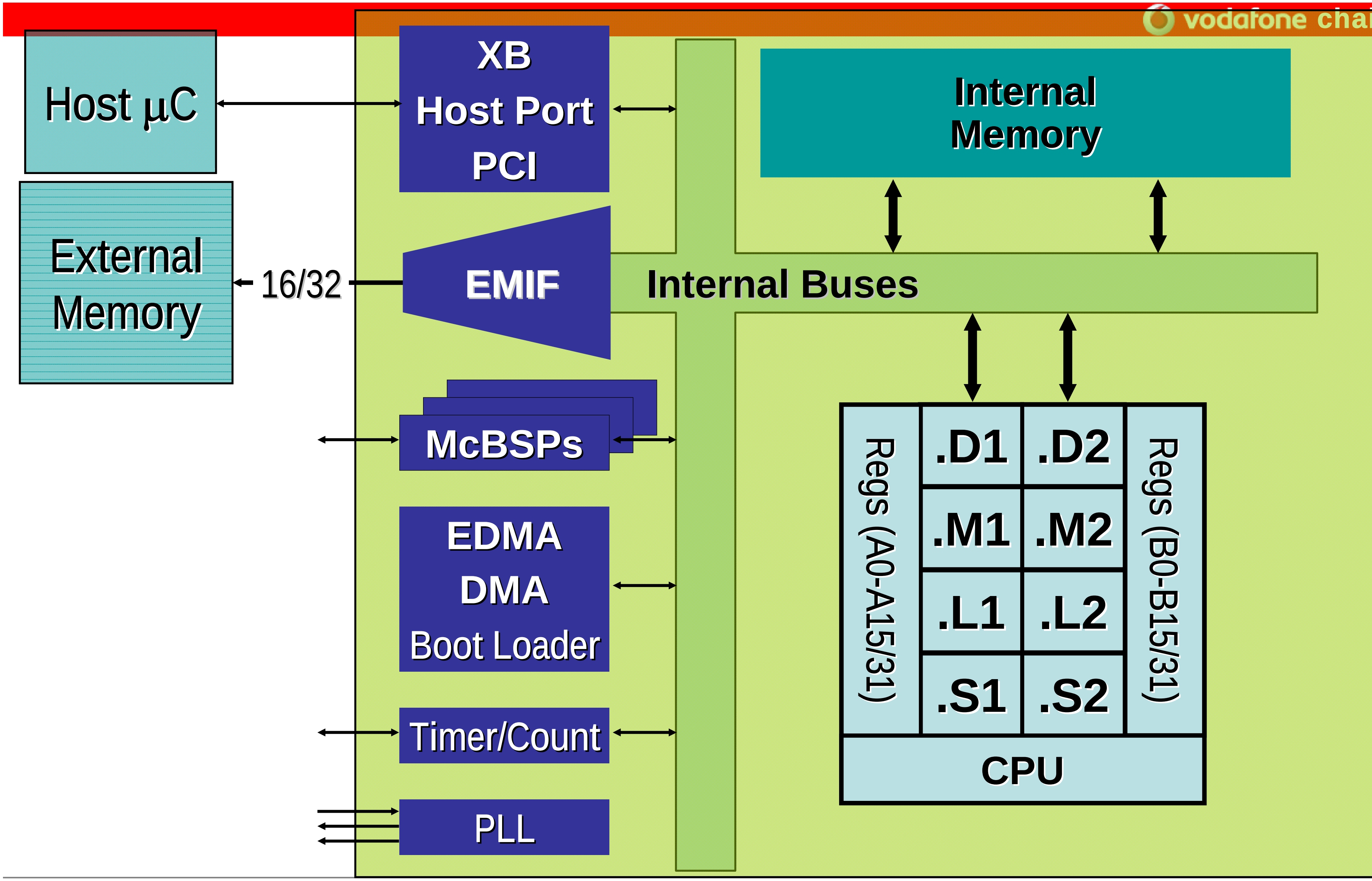
$$a * b = (a_h << 16 + a_l) * (b_h << 16 + b_l)$$

$$= [(a_h * b_h) << 32] + [(a_l * b_h) << 16] +$$

$$[(a_h * b_l) << 16] + [a_l * b_l]$$

Selected ‘6711 Peripherals

C6000 Peripherals



The McBSP

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

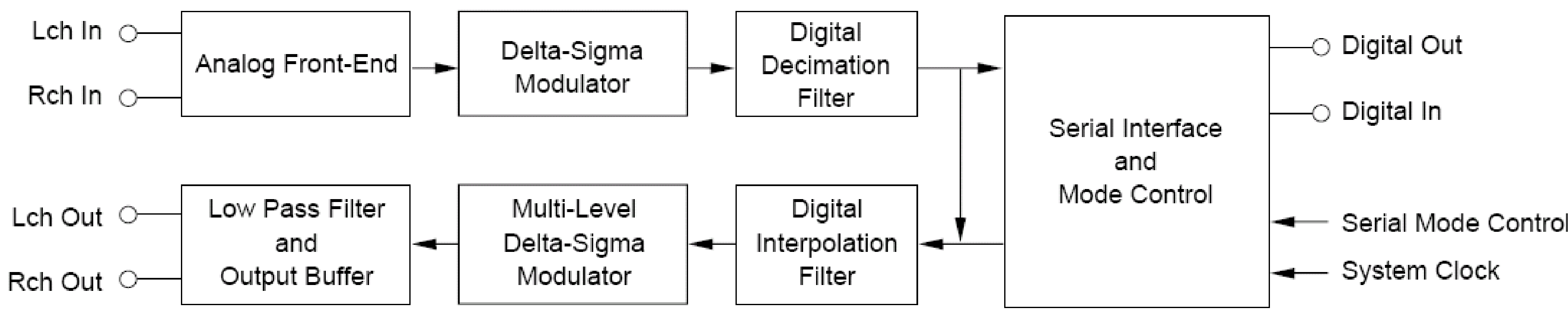
- Multichannel Buffered Serial Port
- Up to **100 Mb/sec** performance
- 2 (or 3) full-duplex, synchronous serial-ports
- Enables **direct interfacing** to industry standard Codecs, Analog interface Chips and other serially connected devices
- Supports a wide range of **data-sizes**, including 8, 12, 16, 20, 24 and 32 bits

  ➔Bit, Word(channel), Frame, Phase

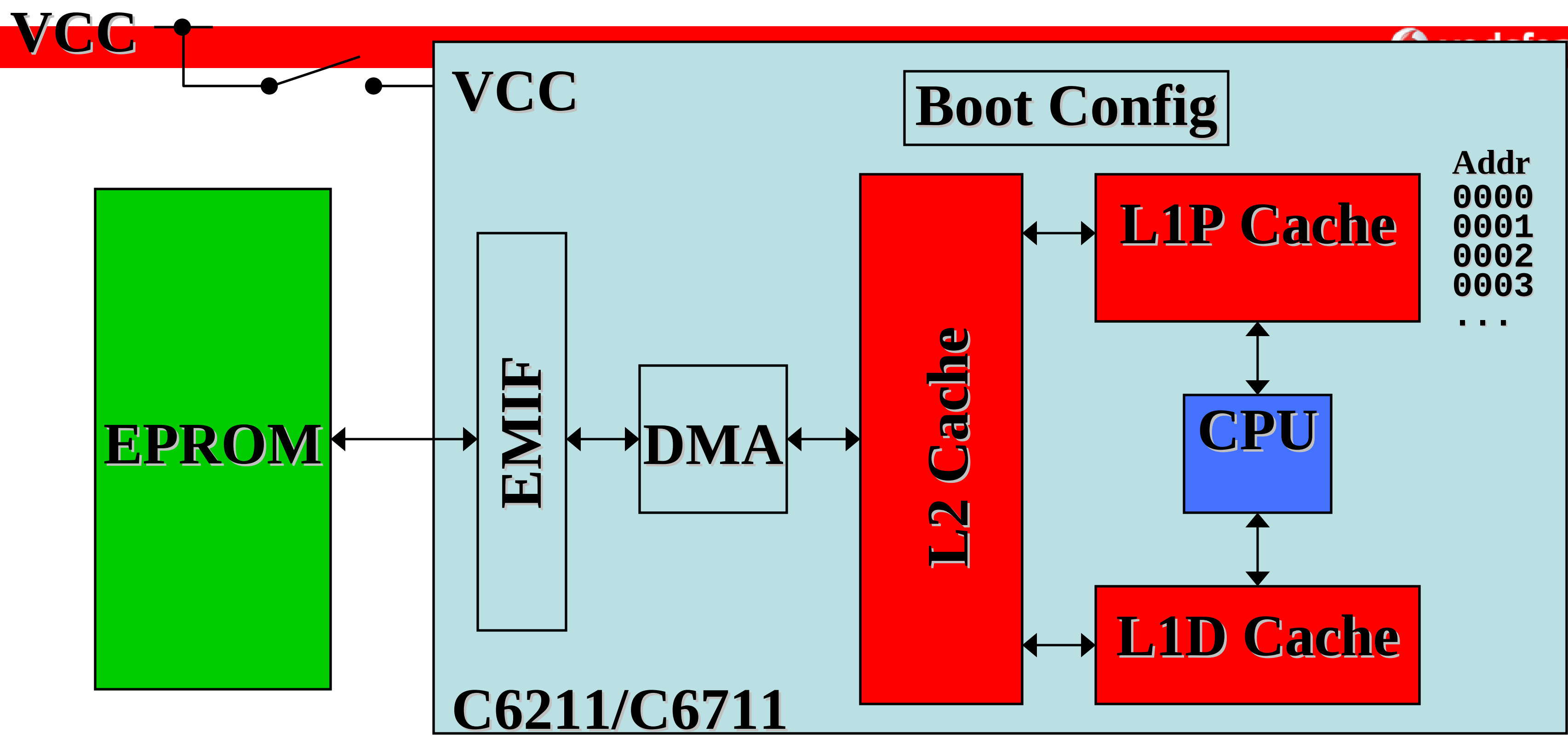
➔In our lab the McBSP is used to connect to the A/D, D/A daughter card

PCM3003

- MONOLITHIC 20-BIT DS ADC AND DAC
  - 16-/20-BIT INPUT/OUTPUT DATA
  - HARDWARE CONTROL: PCM3003
  - STEREO ADC: SNR: 90dB & DynamicRange: 90dB
  - STEREO DAC: SNR: 94dB & Dynamic Range: 94dB
  - Digital Attenuation (256 Steps), Soft Mute, Digital Loop Back
- SAMPLING RATE: Up to 48kHz
- SYSTEM CLOCK: 256f_S, 384f_S, 512f_S

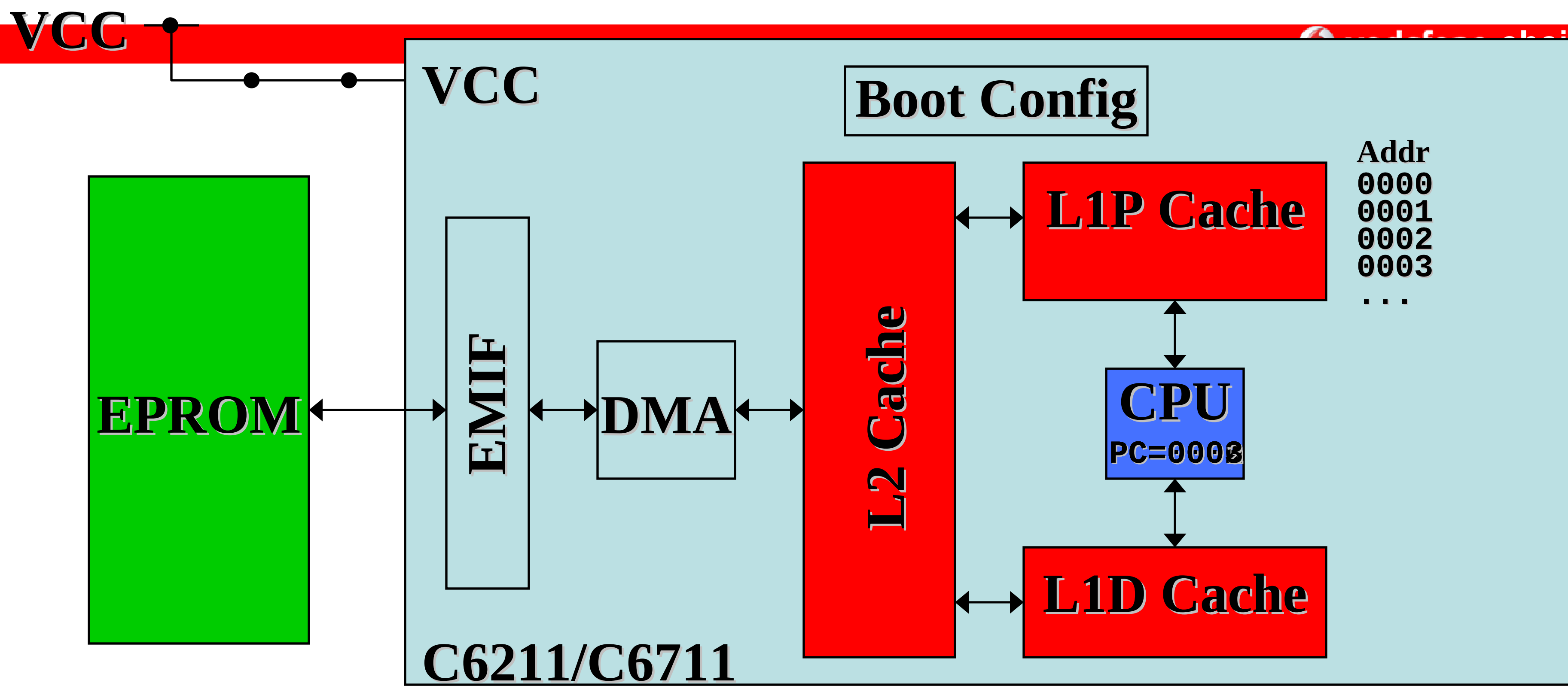


What is the bootloader?



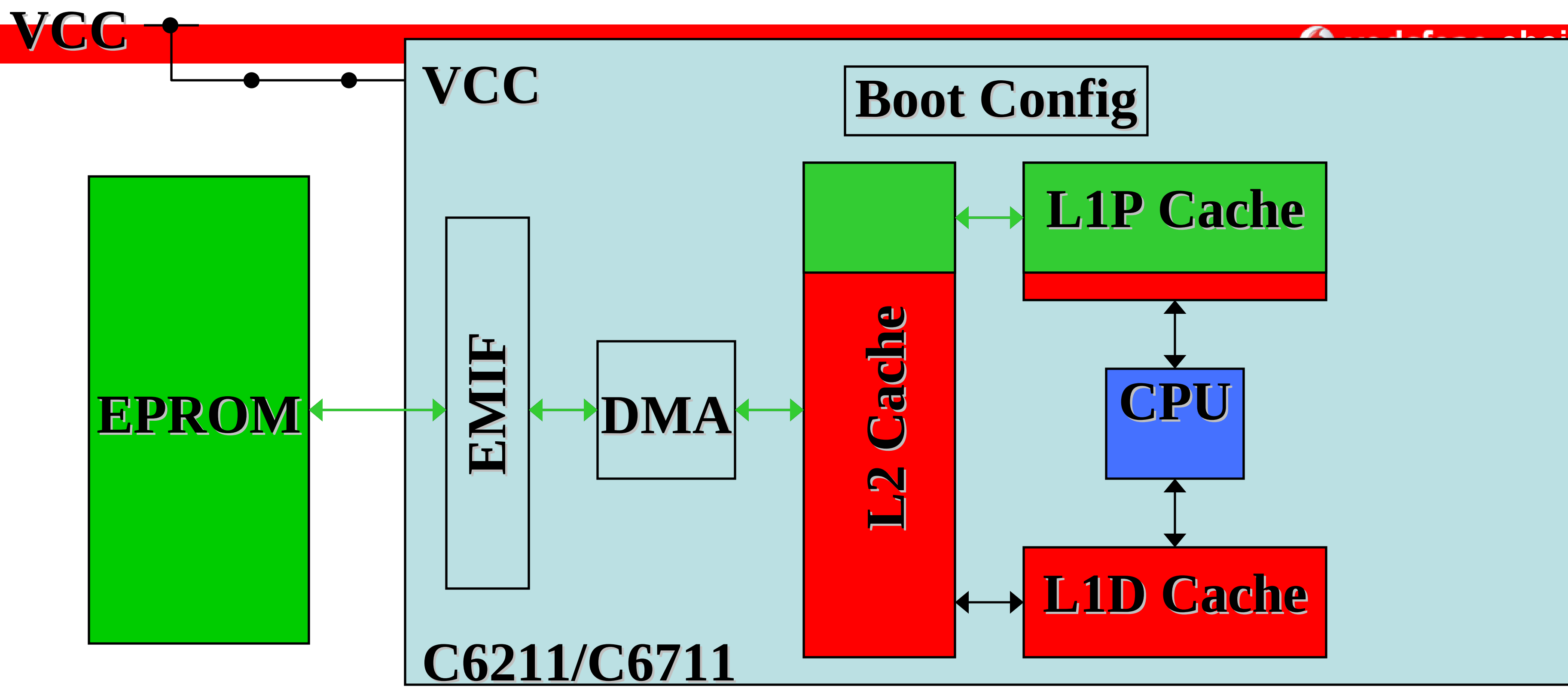
- When the DSP is NOT powered or under reset the internal program memory is in a random state.

What is the bootloader?



- When the DSP is powered and the CPU is taken out of reset the internal memory is still in a random state and the program will start running for address zero.

What is the bootloader?



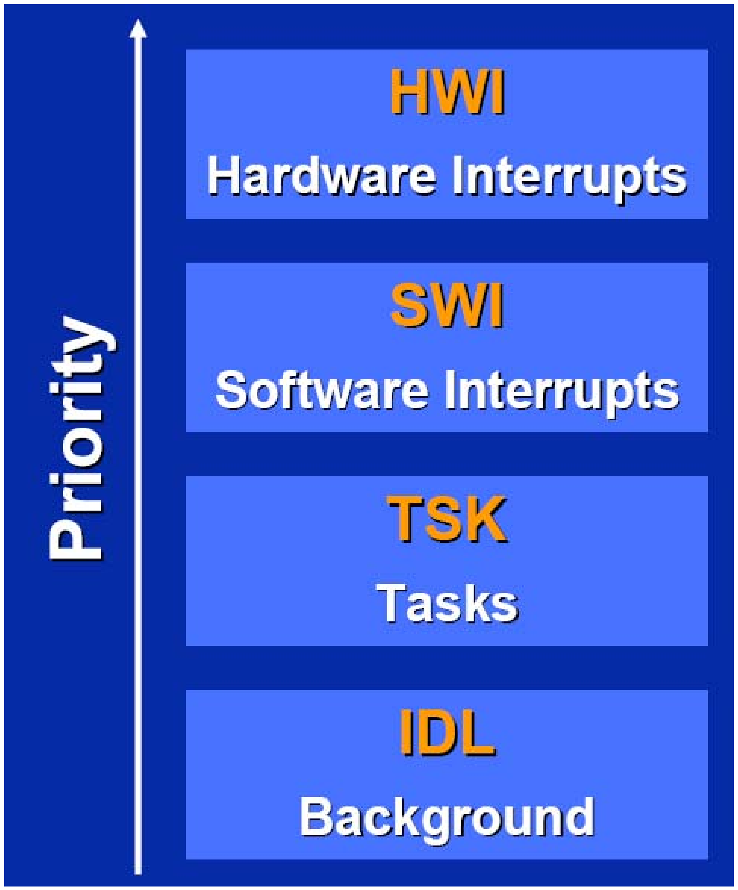
- With the boot, a portion of code can be automatically copied from external to internal memory.

Interrupts

- DSPs must be able to execute tasks on **asynchronous events**
- Interrupts suspend the current processor task and save its context
- A **interrupt service routine** (ISR) is executed
- After completion of the ISR, the context of the former task is restored and the execution continues
- Interrupts are organized hierarchically

➔ vs. Polling

Interrupt Interrupt- and Thread Types



- HWI priorities set by hardware
  ➔One ISR per interrupt
- 14 SWI priority levels ➔Multiple SWIs at each level
- 15 TSK priority levels ➔Multiple TSKs at each level
- Multiple IDL functions
  ➔Continuous loop

➔HWI triggered by hardware interrupt

➔IDL runs as the background thread

TECHNISCHE UNIVERSITÄT DRESDEN
vodafone chair

The DSK6711 Development Kit

DSK Contents

TECHNISCHE UNIVERSITÄT

Hardware

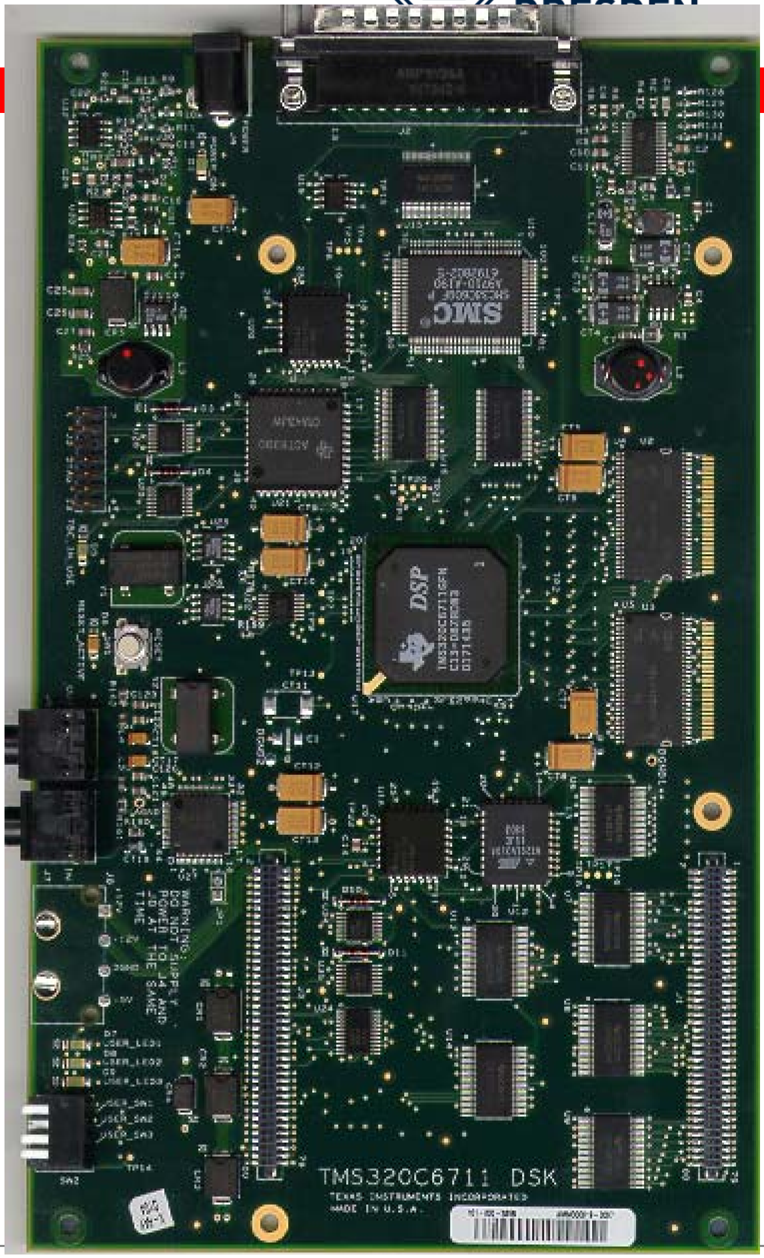
- 150 MHz ‘C6711 DSP
- TI 16-bit A/D Converter (‘AD535)
- External Memory
  - 16M Bytes SDRAM
  - 128K Bytes Flash ROM
- LED’s
- Daughter card expansion
- Power Supply & Parallel Port Cable

Software

- Code Generation Tools

  (C Compiler, Assembler & Linker)
- Code Composer Debugger

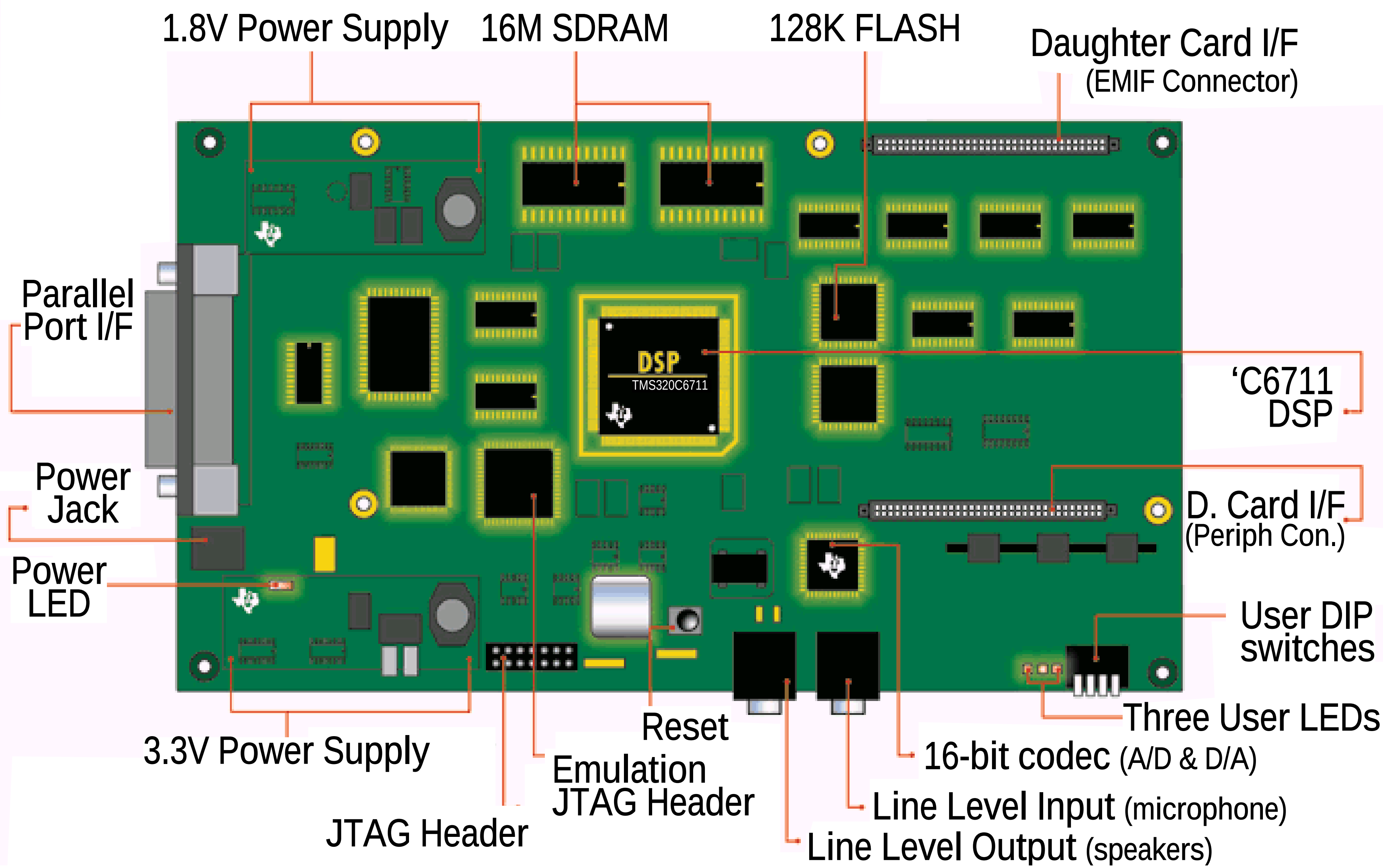
  (256K program limitation)
- Example Programs & S/W Utilities
  - Power-on Self Test
  - Flash Utility Program
  - Board Confidence Test
  - Host access via DLL
  - Sample Program(s)



C6711 DSK Overview

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

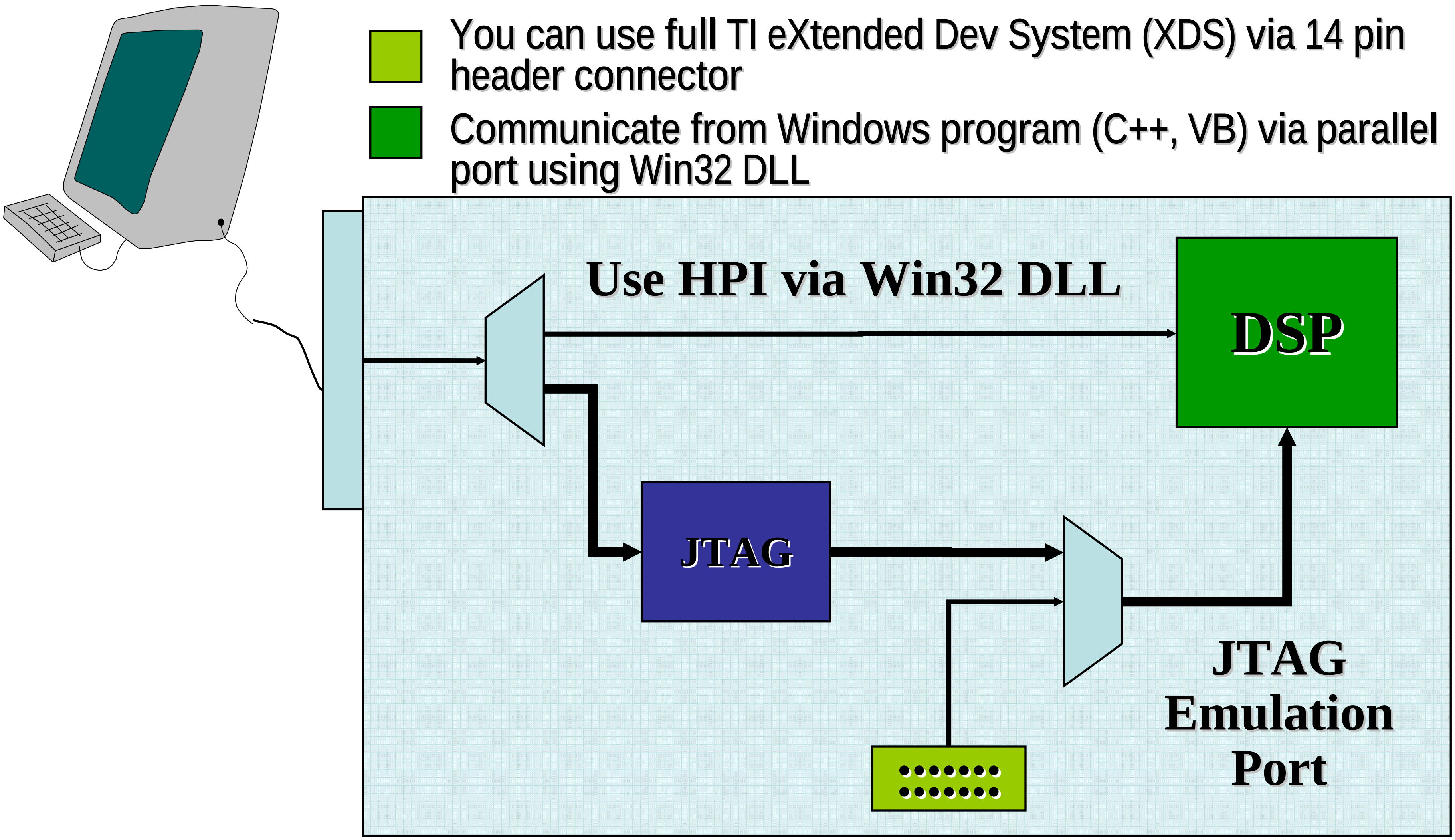


Software: (4) PC → DSK Communications

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

- CCS uses parallel port to control DSP via JTAG port
- You can use full TI eXtended Dev System (XDS) via 14 pin header connector
- Communicate from Windows program (C++, VB) via parallel port using Win32 DLL



What happens to the Source-Code?

